



Statify

Jackson McDaniel
Shannon Marquiss
Joshua Boe
Casey Daly
Brian Ishii

Table of Contents

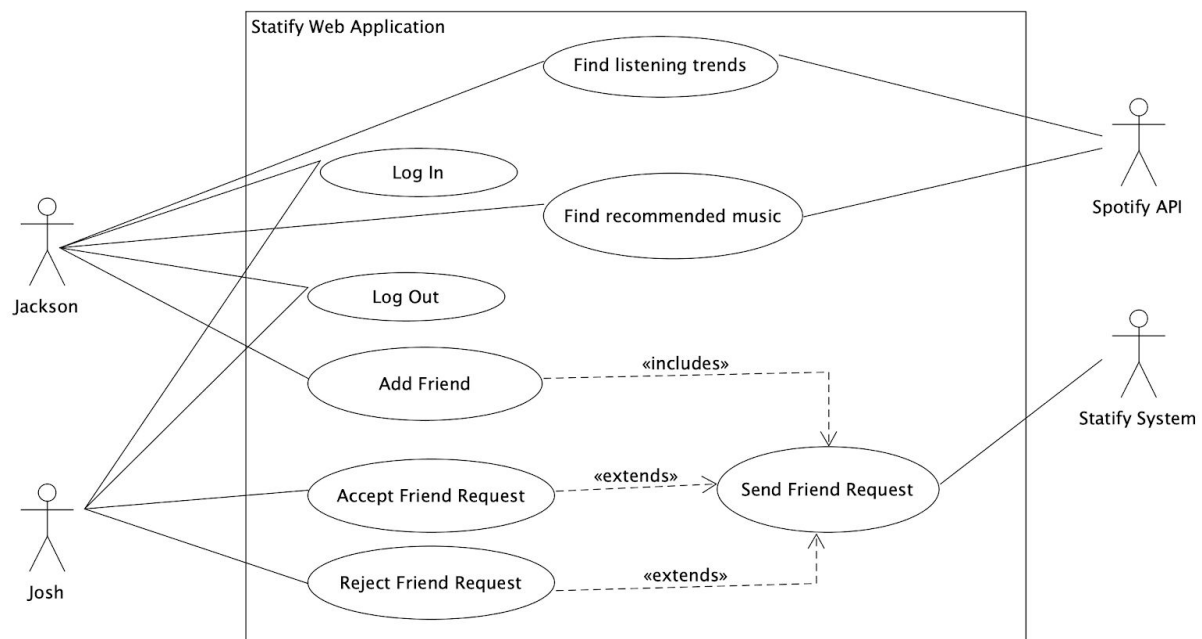
Table of Contents	2
System Description	3
User and System Requirements	3
User Requirement 1 - Log In	3
User Requirement 2 - Log Out	4
User Requirement 3 - Homepage	4
User Requirement 4 - Personal Listening Trends	5
User Requirement 5 - Adding Friends	5
User Requirement 6 - Friends' Trends	5
User Requirement 7 - Recommended Music	6
Non-functional System Requirements	6
User Stories and Acceptance Criteria	6
User Story 1 - Delete Account	6
User Story 2 - Working Statify Web Application	6
User Story 3 - Gather User Data Via Spotify API	7
User Story 4 - Change Account Information	7
User Story 5 - Navigation Bar	7
User Story 6 - User Feedback	7
Risk Driven Specifications and Security Requirements	7
Risk 1 - Unauthorized access of user information	7
Security Requirements due to Risk 1	8
Risk 2 - SQL Injection Attack via User Login	8
Security Requirements due to Risk 2	9
Risk 3 - Maintenance of Database	9
Security Requirements due to Risk 3	9
List of Security Requirements	9
Glossary	10

System Description

Statify is a web app that will pull data from Spotify to give listening insights to its users. Users will be able to login with Spotify, giving us access to Spotify's API endpoints to gather information about the user's listening habits. After gathering and curating this data, we can highlight interesting listening trends based on different factors such as location, genre, age, and more. In addition, we plan on integrating machine learning algorithms to suggest new music for users to listen to.

High level functionalities of the system:

- Users login with their Spotify account
- Show listening trends of individual and groups of users
- Recommend new music to users based on past listening trends
- Show user's friends listening trends and new recommended music



User and System Requirements

User Requirement 1 - Log In

User Requirement: Statify shall allow users to login using their Spotify account.

System Requirements:

1. Statify shall offer a login page where a user can login with their Spotify account information. (functional)
2. Statify shall also display a "Sign up" button if the user has not already signed up for Statify (functional)
3. Statify shall authenticate the Spotify user so that Statify has access to Spotify's API endpoints which require authorization in order to deliver information. (functional)
4. Statify shall cancel a new user login attempt and display a "User already exists" message if there is already an existing user in our database associated with that username/email. (functional)
5. Statify shall add new users to the database and allocate space for data associated with them. (functional)
6. Statify shall require user to have a Spotify account to use the web application. (nonfunctional)
7. Statify shall ask for consent from user to log in through Spotify account. (nonfunctional)

User Requirement 2 - Log Out

User Requirement: Statify shall allow users to log out of their Statify account.

System Requirements:

1. Statify shall offer a clickable "Log Out" button in the navigation bar. This button will be displayed on all pages aside from the log-in page as long as the user is logged in. (functional)
2. When the "Log Out" button is clicked, Statify shall display a dropdown table that asks the user to confirm logging out by displaying "Log Out?" with "Yes" and "No" options. (functional)
3. When "Yes" is selected from the confirmation dropdown, the user shall be logged out and redirected to the login page. (functional)
4. When "No" is selected from the confirmation dropdown, the dropdown shall close and the user will remain on the current page. (functional)
5. Statify shall close the session only if the user closes the tab or manually logs out. (nonfunctional)

User Requirement 3 - Homepage

User Requirement: Statify shall display a homepage to the users upon verification of their login information.

System Requirements:

1. Statify's homepage shall display a navigation bar where the user can choose to go to one of the pages on the web site. (functional)

2. Statify's homepage shall display general user listening trends for all Statify users.
(functional)

User Requirement 4 - Personal Listening Trends

User Requirement: Statify shall allow users to view their listening trends based on certain time intervals.

System Requirements:

1. Statify shall display listening trends in the form of top songs, artists, and playlists
(functional)
2. Statify shall display current trend information of all Spotify users on the homepage.
(functional)
3. Statify shall display to each user their own personal listening trend information.
(functional)
4. Statify shall store listening trend information in its database so the user may access it later. (functional)
5. Statify shall serve stored listening trend information at the request of the user.
(functional)

User Requirement 5 - Adding Friends

User Requirement: Statify shall allow users to add friends on Statify.

System Requirements:

1. Statify shall allow users to search for friends using the navigation bar. (functional)
2. Statify shall display an "Add Friend" button on another user's public page. (functional)
3. Statify shall send an e-mail to a user when they have a friend request. (functional)
4. Statify shall allow users to accept or deny friend requests sent to them. (functional)
5. Statify shall allow a user to view a friend's private page once the other user has accepted the request. (functional)

User Requirement 6 - Friends' Trends

User Requirement: Statify shall allow users to view their friends music libraries and listening trends.

System Requirements:

1. Statify shall have a search bar where users can type in their friend's names. (functional)
2. Statify shall provide a means of filtering down the user's friends based on what the user types into the search bar. (functional)

3. Statify shall allow the user to click on the names of their friends in the search bar and bring the user to their friends Statify account pages. (functional)
4. Statify shall display the friend's listening trends, playlists, and music library on this page. (functional)
5. Statify shall allow the friend's listening trends to be filtered over a chosen time interval. (functional)
6. Statify shall display songs recommended to their friends. (functional)

User Requirement 7 - Recommended Music

User Requirement: Statify shall allow users to view recommended music based on their previous listening trends.

System Requirements:

1. Statify shall have a page dedicated to music recommended to the user. (functional)
2. Statify shall list the music previously listened to by the user that determines which music is recommended. (functional)
3. Statify shall tune its recommendations based on the recommended songs that the user chooses to listen to. (functional)

Non-functional System Requirements

1. Statify shall need SQL as a database.
2. Statify shall be hosted on AWS.

User Stories and Acceptance Criteria

User Story 1 - Delete Account

As a user, I can delete my account if I no longer want to be associated with my data that is being shared to Statify.

1. On the user's profile page, the current account information is displayed.
2. A "Delete Account" button is available and displayed within the current account information.
3. Statify no longer grabs data once the user deletes their account.
4. Any personally Identifiable information shall be removed from the database.
5. User receives confirmation (email) to verify that they deleted their account.

User Story 2 - Working Statify Web Application

As a user, I want to be able to open Statify as a web application.

1. Statify is supported on Google Chrome, Firefox, and Safari.

User Story 3 - Gather User Data Via Spotify API

As a developer, I want to be have users' data taken from Spotify's API and then parsed and organized for use.

1. Data is read through Spotify's API endpoint.
2. Data is parsed and organized so that it can be used in Statify.

User Story 4 - Change Account Information

As a user, I want to be able to change my Statify account information.

1. Statify has a specific page dedicated to changing account information and toggling user settings.
2. Users can access this from the navigation bar.
3. Every field of information will be filled in with their current account information.
4. Users can click to edit this information
5. Users can click a "save" button to save their information before they leave the page.

User Story 5 - Navigation Bar

As a user, I want to be able to navigate the web application using a navigation bar.

1. Statify has a navigation bar available on every page in the Statify web application.
2. Users can click the navigation bar to redirect them to other pages on Statify.
3. The navigation bar can redirect a user to the homepage, listening trends, recommended music, and friends' pages.
4. The navigation bar will include a search bar to search for friends.

User Story 6 - User Feedback

As a user, I want to be able to provide feedback to Statify regarding the recommended songs

1. Statify will display a thumbs up and thumbs down image next to songs that a user can press to indicate whether they like or dislike the song.
2. Statify will use this data to make recommendations in the future according to these responses.

Risk Driven Specifications and Security Requirements

Risk 1 - Unauthorized access of user information

The System shall not allow unauthorized access of user information

Risk Identification

The risk is that unauthorized users can see the listening trends and recommendations of Statify users without their permission. The unauthorized users would also then have access to the user's account information.

Risk Analysis

- The risk is a breach of privacy, which needs to be prevented. The risk of malicious users gaining access to user information has the potential to be high because there may be sensitive information stored about each user, depending on what information each user chooses to share with Statify.

Risk Decomposition

- If someone guessed or found out someone's password
- If our database is disorganized or secured poorly
- If data associated with an account other than the intended user's is intercepted.

Risk Reduction

- Encryption of data
- Utilize Spotify login for authentication

Security Requirements due to Risk 1

Statify shall not allow unauthorized access of user information

Statify shall not store any unencrypted data on the database

Risk 2 - SQL Injection Attack via User Login

The System shall prevent SQL injection attacks.

Risk Identification

Statify will combine the users' login information with a SQL command in order to store it in our SQL database. The risk is that a malicious user will enter a valid SQL command into the login information text box (such as username or password), which could be used to access unauthorized information or corrupt the data set.

Risk Analysis

- The risk of a SQL Injection attack has a relatively low possibility given that most of the information stored in the database is not security critical. If a malicious user did want to use SQL injection to insert data into the database, it could put what data we do store for users in serious risk.

Risk Decomposition

- Root cause of a SQL injection attack would be formatting user login information to take the form of a SQL command

Risk Reduction

- Require a specific format for usernames that do not allow SQL formatted strings
- Require user login information to be just one word (not allow spaces)
- Use Spotify accounts to log the user in should eliminate most of this risk

Security Requirements due to Risk 2

Statify shall not accept improperly formatted requests to minimize the likelihood of a SQL injection attack.

Risk 3 - Maintenance of Database

The System shall not lose or corrupt user data.

Risk Identification

The risk is that user's data can be lost or corrupted. This can cause accounts to be lost, and important data being misinterpreted.

Risk Analysis

- The risk of user data being lost or corrupted is medium risk. The amount of data and correctness of data is very important for creating good recommendations. The risk is if our server crashes and does not have viable backups.

Risk Decomposition

- Server crashes due to power outage or other issue
- Software bugs overwrite data stored
- Server failures

Risk Reduction

- Data redundancy on backups

Security Requirements due to Risk 3

Statify shall not backup data less than one time a day

List of Security Requirements

- Statify shall not allow unauthorized access of user information
- Statify shall not store any unencrypted data on the database
- Statify shall not accept a request without having a captcha verify a human user
- Statify shall not accept improperly formatted requests to minimize the likelihood of a SQL injection attack.
- Statify shall not backup data less than one time a day

Glossary

API Endpoint - A unique URL that represents an object or collection of objects. (i.e. a user's top artists or tracks)

Captcha - a program or system intended to distinguish human from machine input, typically as a way of thwarting spam and automated extraction of data from websites

Encryption - Formatting data in a way such that it would be unreadable to anyone with possible malicious intent.

Library - All songs saved by a user on the Spotify app.

Listening trend - Music that a user has been listening to over the most recent specified amount of time, and can be filtered by artists, songs, and genres.

Navigation Bar - A bar that assists in navigating a site by displaying clickable links that redirect the user to the destination at which they clicked.

Playlist - A collection of songs grouped together by a user on the Spotify app.